

Práctica Universitaria Conducente a Trabajo de grado. Grupo Investigación GIA

Practicante:

Juan Camilo Chavarro Acosta

Director de la práctica:

Julio César López Betancur

Universidad Tecnológica De Pereira Facultad De

Ingenierías

Programa Ingeniería De Sistemas Y Computación

Pereira - Colombia

2021

Índice

Resumen	4
Abstract	4
Introducción	5
Objetivos	6
Objetivo General	6
Objetivos Específicos	6
Planteamiento Del Problema	7
Justificación	8
Marco Teórico Y Conceptual	9
Marco Metodológico	11
Proceso De Ingeniería De Software	12
Bibliografía	40

Índice de figuras

Figura 1. módulo de OAuthClientDetailsController	12
Figura 2. Módulo de PermissionCategoryController	13
Figura 3. Módulo de PermissionController	13
Figura 4. RolController	14
Figura 5. UserController	14
Figura 6. Casos de uso HIS	15
Figura 7. Casos de uso Bodega	15
Figura 8. Casos de uso Consulta	16
Figura 9. Casos de uso Transformador	16
Figura 10. Casos de uso PosicionControler1	17
Figura 11. Casos de uso PosicionControler	18
Figura 12. Casos de uso ConectorControler	18
Figura 13. Casos de uso ComponenteControler	18
Figura 14. Casos de uso CategoriaControler	19
Figura 15. Casos de uso Diligenciamiento	20
Figura 16. Casos de uso Formulador	22
Figura 17. Casos de uso resultados	22
Figura 18. Casos de uso indicadores	23
Figura 19. Casos de uso RecursoTotalSede	24
Figura 20. Casos de uso Propuesta	24
Figura 21. Casos de uso PropuestaCheck	25
Figura 22. Casos de uso Demanda	26
Figura 23. Casos de uso AnalisisPropuesta	26
Figura 24. Casos de uso Actualizar	26
Figura 25. Casos de uso TalentoHumanoControler	27
Figura 26. Casos de uso Servicio	28

Figura 27. Casos de uso RecursoFisico	28
Figura 28. Casos de uso Modalidad	29
Figura 29. Casos de uso IPS	30
Figura 30. Casos de uso IPS	31
Figura 31. Casos de uso Crupo	31
Figura 32. Casos de uso Departamento	32
Figura 33. Casos de uso Complejidad	32
Figura 34. Casos de uso Análisis	32
Figura 35. Casos de uso Administrador	33
Figura 36. Casos de uso Microterritorios	34
Figura 37. Casos de uso Departamento	34
Figura 38. Casos de uso Consultar	35
Figura 39. Casos de uso Actualizar	35
Figura 40. Casos de uso Posición	35
Figura 41. Casos de uso RiasConfiguracion	36
Figura 42. Casos de uso Paciente	36
Figura 43. Casos de uso Historial	37
Figura 44. Casos de uso Ejecución	37
Figura 45. Casos de uso RiasConfiguracion	38
Figura 46. Casos de uso MonitoreoRias	38
Figura 47. Casos de uso MonitoreoComponente	39

Resumen

El presente documento tiene como finalidad documentar el proceso llevado a cabo durante las prácticas universitarias, en la cual se desarrollaron dos tipos de actividades, en el primer mes, tuvo lugar una capacitación en el área de desarrollo en las herramientas Spring Boot y Angular, herramientas usadas para el desarrollo backend y frontend, también herramientas como Git, GitHub, Markdown y Plantuml.

En los siguientes meses de práctica, se documentaron dos proyectos, los cuales fueron Bus de servicios de salud e interoperabilidad en salud.

La documentación de los proyectos contiene los siguientes puntos: README, diagramas de clases, diagramas de casos de uso, escenarios de actividades y diagramas de secuencia.

PALABRAS CLAVE: Desarrollo, backend, frontend, documentación, proyecto, Git, Github, EndPoint, Markdown, Plantuml.

Abstract

The purpose of this document is to document the process carried out during the university practices, in which two types of activities were developed, in the first month, a training took place in the area of development in the Spring Boot and Angular tools, tools used for backend and frontend development, also tools like Git, GitHub, Markdown and Plantuml.

In the following months of practice, two projects were documented, which were Health Services Bus and Health Interoperability.

The project documentation contains the following items: README, class diagrams, use case diagrams, activity scenarios and sequence diagrams.

KEYWORDS: Development, backend, frontend, documentation, project, Git, Github, Markdown, Plantuml.

1. Introducción

La práctica Universitaria Empresarial está enfocada a afianzar los conceptos de la ingeniería del software, desarrollo, soporte y mantenimiento del software en el campo del desarrollo web, el cual es el foco de atención de los procesos en los proyectos del grupo de investigación GIA UTP, de acuerdo a las líneas de investigación y extensión que se trabajan en el grupo de investigación.

2. Objetivos

2.1 Objetivo General

Participar en el desarrollo, soporte y mantenimiento de aplicativos que requiere el grupo de Investigación para el cumplimiento de sus objetivos académicos en investigación y extensión

2.2 Objetivos Específicos

Capacitación en herramientas de desarrollo

Aplicación de la ingeniería de software en la práctica Universitaria

Aplicación de metodologías ágiles en el proceso de desarrollo de software

3. Planteamiento Del Problema

El grupo de investigación GIA UTP, el cual se encuentra adscrito al programa de Ingeniería de sistemas y computación, adelantó líneas de trabajo en investigación y extensión, donde constantemente requiere la aplicación de la ingeniería de software, para el diseño, pruebas y documentación para soporte. También requiere metodologías para el desarrollo de software, como por ejemplo la metodología SCRUM, planeación de proyectos y administración de proyectos de software para cumplir de la mejor manera con los objetos contractuales con las entidades con las cuales tiene convenios.

4. Justificación

En el desarrollo de software existen distintas metodologías y etapas para llevar a cabo un proyecto, etapas como el planteamiento del problema, planteamiento de la solución, diseño del software; que será la solución o irá de la mano de esta, implementación del desarrollo de software, pruebas al programa, corrección de errores al software y documentación de soporte. Al tener tantas etapas o actividades a desarrollar, es evidente que se necesita una metodología ágil, como SCRUM y un equipo de trabajo completo y acorde a las necesidades del proyecto. Por esto, en el grupo de investigación GIA existe un grupo de trabajo robusto, con personas enfocadas a las distintas etapas mencionadas, precisamente en la de documentación de soporte y diseño, siendo las áreas en las cuales se desempeñó mayormente la práctica universitaria.

La necesidad de una correcta documentación al final de proyecto de software es alta, siendo de gran importancia para el soporte de la aplicación, como también para una correcta mantenibilidad, de esta manera brindando facilidad a los desarrolladores venideros en el proyecto de software al momento de entender, mejorar, corregir errores, depurar el código y de seguir desarrollando el programa. Claramente todas las etapas del desarrollo de software son sumamente importantes, sin excepción de la anteriormente mencionada.

5. Marco Teórico Y Conceptual

La ingeniería de software, a diferencia de la ingeniería de sistemas, está enfocada en el diseño, instrumentos y procesos usados para el desarrollo del software, haciéndola una sola rama de la ingeniería de sistemas de software y computación. Dejando claro de esta manera que la ingeniería del software es la herramienta concisa que se puede emplear al realizar el diseño de los programas, esto para implementar correctamente los desarrollos de software y seguidamente, realizar una adecuada documentación.

El lenguaje en el cual la ingeniería de software se basa para realizar los diseños de programas y documentación es llamado UML (Unified Modeling Language), el cual contiene diversas formas de describir y diseñar el software por medio de diagramas, de esta manera, cualquier ingeniero del software puede identificar el lenguaje, entender los diagramas e implementar el desarrollo de software descrito en ellos.

La implementación de los programas se puede facilitar por medio de los frameworks, ya que contienen una serie de soluciones similares a las necesarias en los procesos de desarrollo comunes. De esta manera, se puede agilizar correctamente el desarrollo de software, abarcando buenas prácticas de programación y algoritmos optimizados por el mismo framework.

Seguidamente se presentarán los conceptos necesarios para describir el proceso realizado en la práctica universitaria.

UML: El lenguaje de modelado unificado (UML) es un estándar para la representación visual de objetos, estados y procesos dentro de un sistema. Por un lado, el lenguaje de modelado puede servir de modelo para un proyecto y garantizar así una arquitectura de información estructurada; por el otro, ayuda a los desarrolladores a presentar la descripción del sistema de una manera que sea comprensible para quienes están fuera del campo. (Ionos, 2018)

Framework: Un Framework es una herramienta que contiene un conjunto de librerías orientadas a la reutilización de componentes de Software para el desarrollo rápido de aplicaciones y podemos adaptarlo a las necesidades de nuestros proyectos, dando la ventaja de no partir de cero sino más bien podemos hacer uso del código ya existente. (DLRJAD, 2015).

Scrum: Scrum es un proceso de gestión que reduce la complejidad en el desarrollo de productos para satisfacer las necesidades de los clientes. La gerencia y los equipos de Scrum trabajan juntos alrededor de requisitos y tecnologías para entregar productos funcionando de manera incremental usando el empirismo. (Huambachano, 2017)

6. Marco Metodológico

Se aplicó el método ágil Scrum para desarrollar las actividades llevadas a cabo en la práctica universitaria, haciendo más eficaz la capacitación en el framework Sprint Boot y Angular, y la documentación de los proyectos de Interoperabilidad en salud de microterritorios y Bus de servicios de salud.

Dentro de la documentación de los proyectos se tomó como herramienta los diagramas descritos por el lenguaje UML, como lo son los diagramas de clase, diagramas de actividades, diagramas de secuencia, escenarios de actividades.

7. Proceso De Ingeniería De Software

En el grupo de investigación GIA se desarrolló en un tiempo de cuatro meses, el soporte, documentación y mantenimiento de diferentes proyectos, los cuales se distribuyeron entre los practicantes, en total cuatro. Dichos proyectos se presentaron y enseñaron a los practicantes para quedar en competencia de llevar a cabo las actividades anteriormente mencionadas.

Particularmente, se inició con el proceso de reconocimiento del software de Interoperabilidad en salud de microterritorios y Bus de servicios de salud para seguidamente realizar las documentaciones a los distintos módulos del proyecto asignado.

A continuación se describirán los módulos a los cuales se les realizó la documentación de Interoperabilidad en salud de microterritorios y sus actividades con sus respectivos diagramas.

- Módulo de autenticación: Se encarga de autenticar los usuarios en todos los sistemas del proyecto

Diagrama UML de clases de los Controladores para cada “Endpoint”

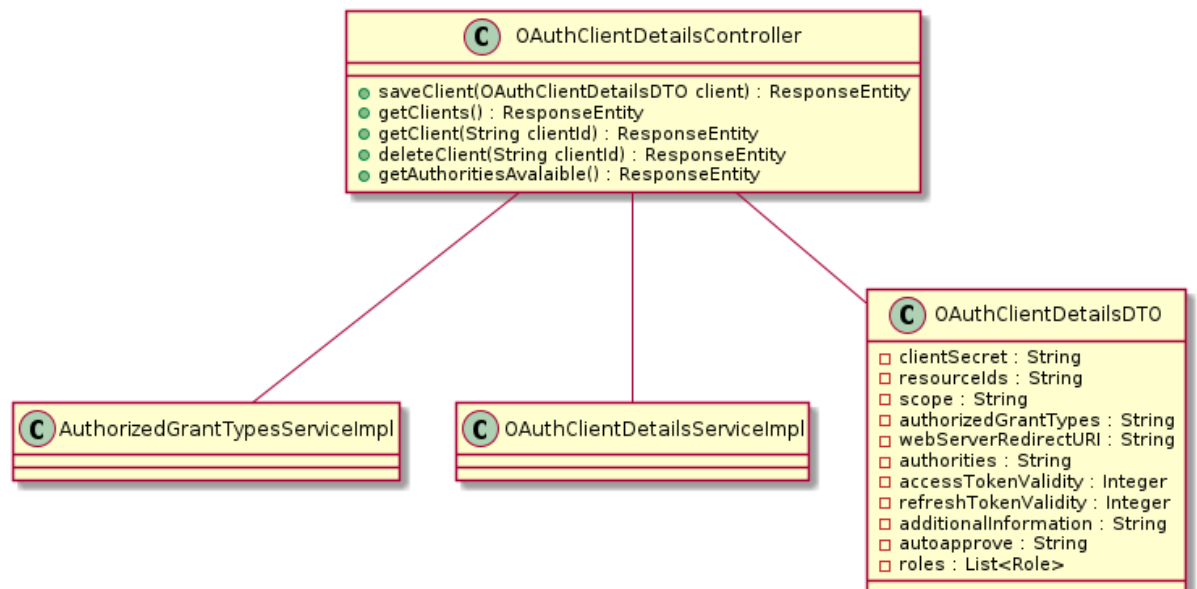


Figura 1. módulo de OAuthClientDetailsController

Este módulo realiza la autenticación en el sistema.

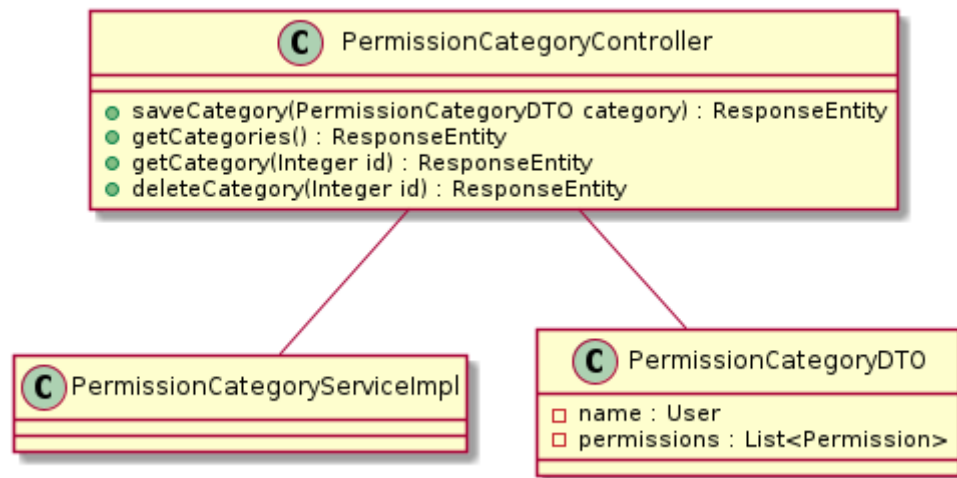


Figura 2. Módulo de PermissionCategoryController

Este módulo se encarga del guardado, la obtención y eliminación de los permisos de categorías .

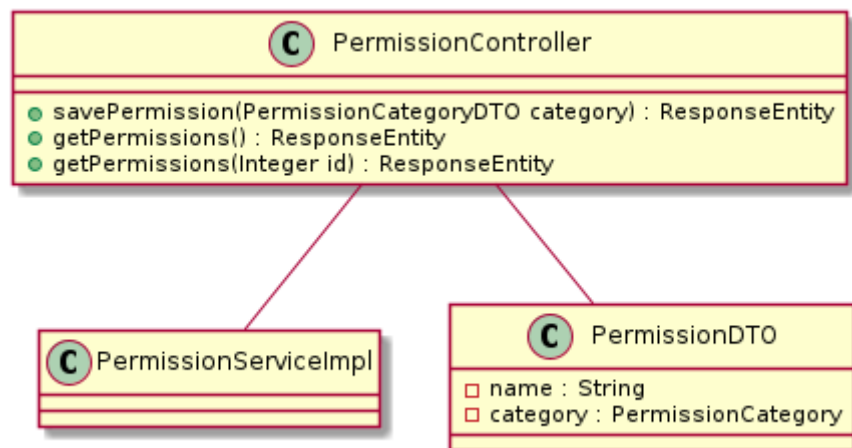


Figura 3. Módulo de PermissionController

Este módulo se encarga del guardado y la obtención de los permisos.

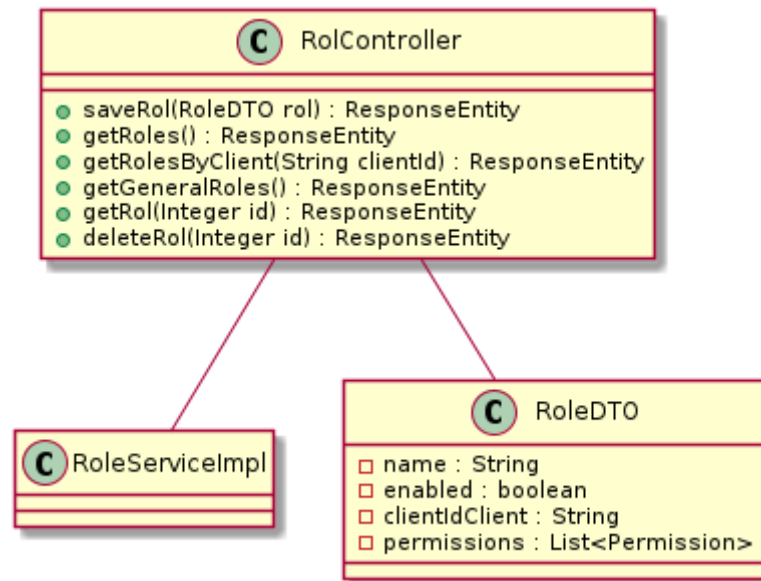


Figura 4. RolController

Este módulo se encarga del CRUD del rol.

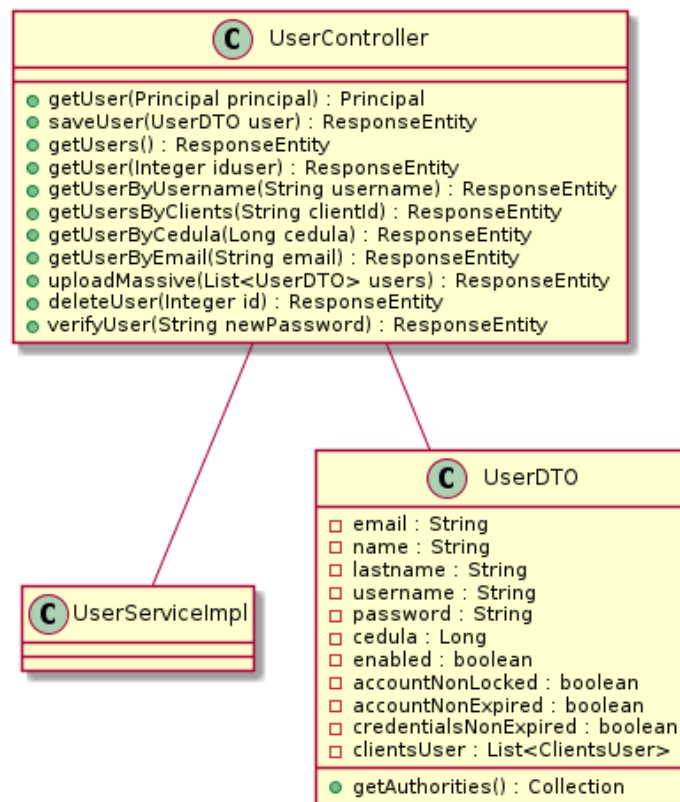


Figura 5. UserController

Este módulo se encarga del CRUD del usuario.

- Módulo de his: Se encarga de la administración del servicio de historias de usuario del sistema

Diagrama UML de Casos de Usos:

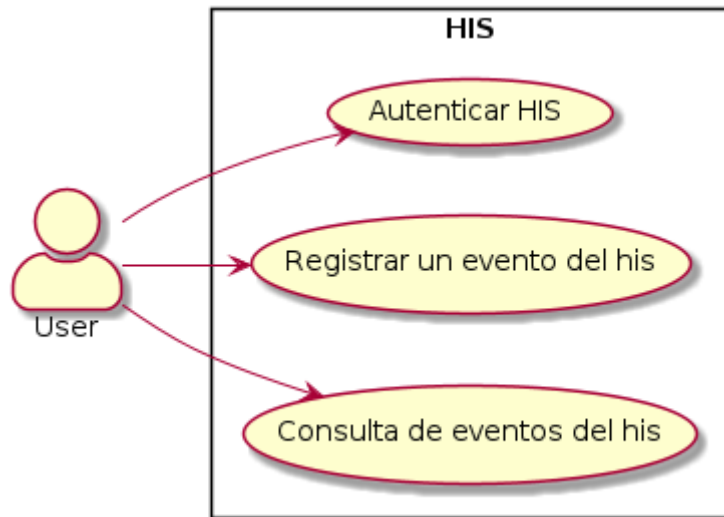


Figura 6. Casos de uso HIS

- Módulo de bodega: Se encarga de la administración de los eventos crudos de usuario, como también el envío de estos al servicio de índices y la transformación de los mismos a la estructura de la bodega de datos.

Diagrama UML de Casos de Usos:

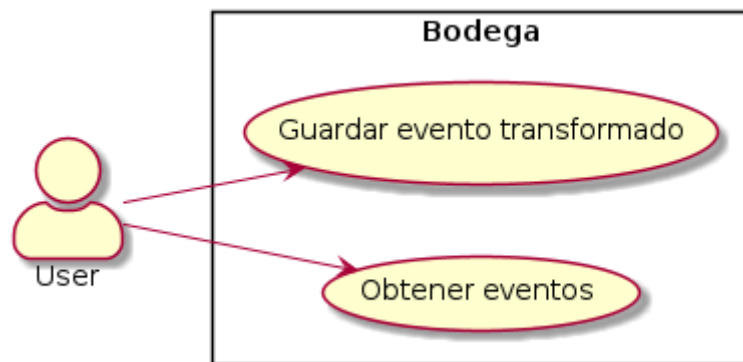


Figura 7. Casos de uso Bodega

- Módulo de consulta: El funcionamiento consiste en la consulta de los eventos de los usuarios.

Diagrama UML de Casos de Usos:

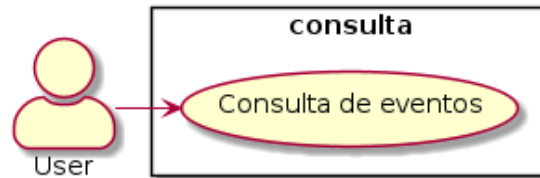


Figura 8. Casos de uso Consulta

- Módulo de recorder (Grabación): Consiste en el registro de las IPS (Instituciones Prestadoras de Salud) por medio de una petición web a sus servicios.
Se omitieron los diagramas para la descripción del módulo por la simpleza del mismo.
- Módulo del transformador: Este módulo se encarga de la transformación de eventos de usuarios desde archivos al sistema.

Diagrama UML de Casos de Usos:

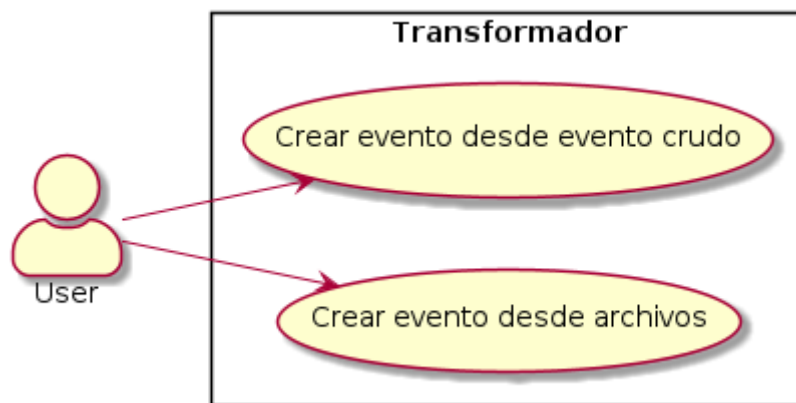


Figura 9. Casos de uso Transformador

A continuación se describirán los módulos a los cuales se les realizó la documentación de Bus de servicios de salud.

- Módulo de configuración: Consiste en el CRUD (Creación, lectura, actualización y borrado de los datos) de encuestas, es decir, gestiona la información del sistema.

Diagrama UML de Casos de Usos:

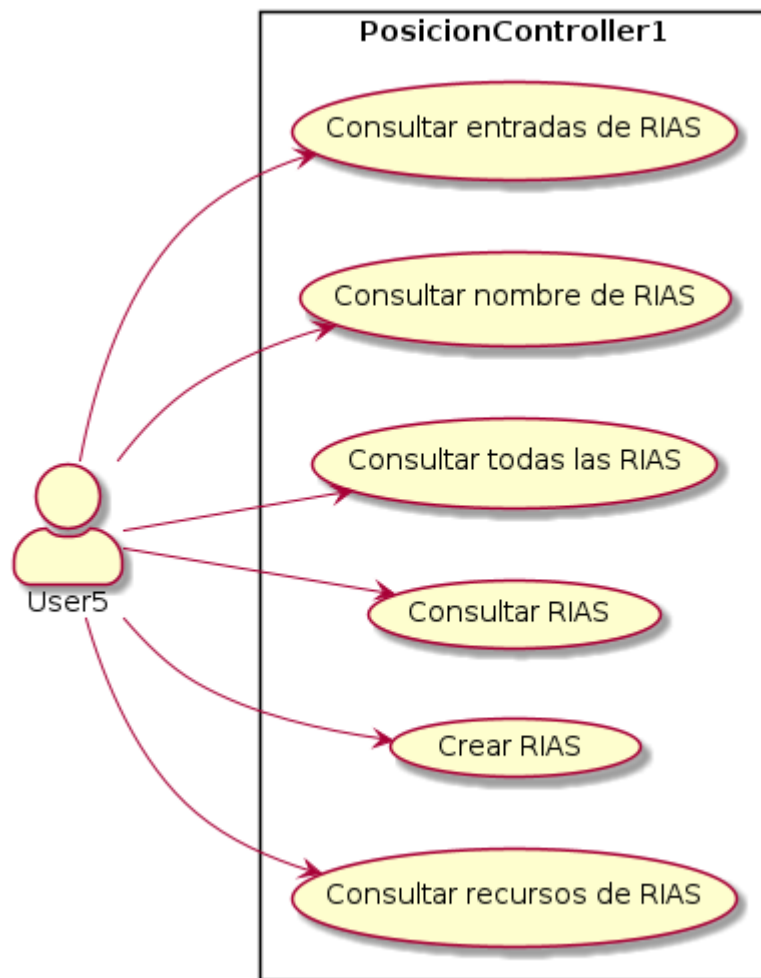


Figura 10. Casos de uso PosicionControler1

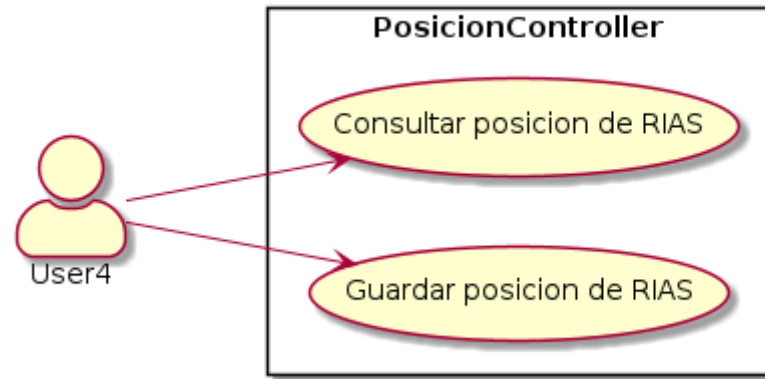


Figura 11. Casos de uso PosicionControler

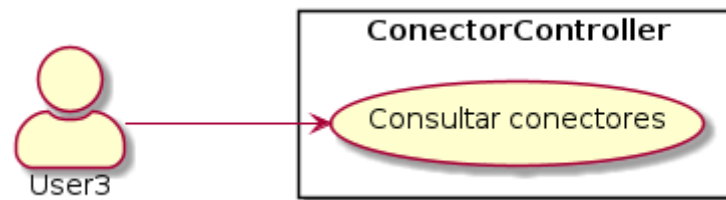


Figura 12. Casos de uso ConectorControler

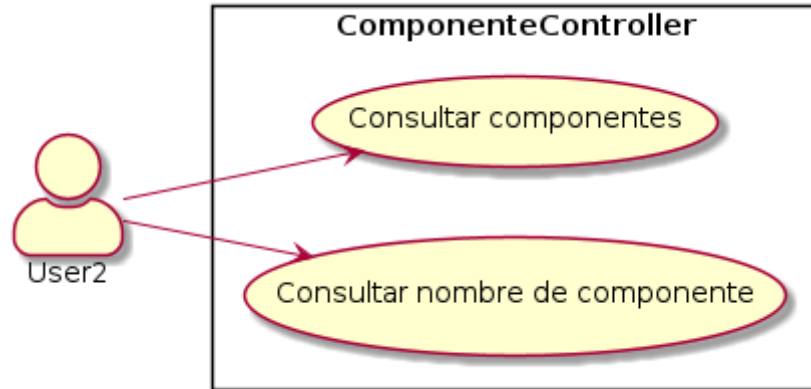


Figura 13. Casos de uso ComponenteControler

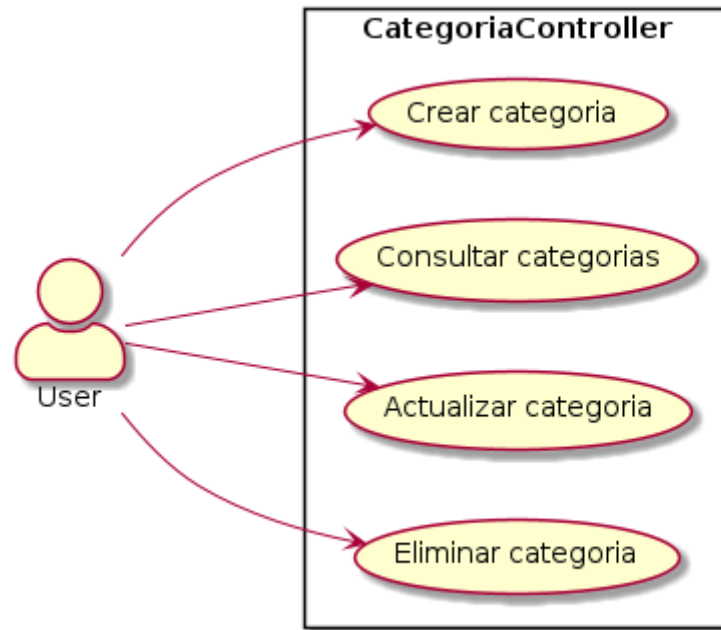


Figura 14. Casos de uso CategoriaController

- Módulo de diligenciamiento: Se encarga de la creación y lectura de los diligenciamientos de encuestas, es decir, gestiona la información que los usuarios ingresan al sistema.

Diagrama UML de Casos de Usos:

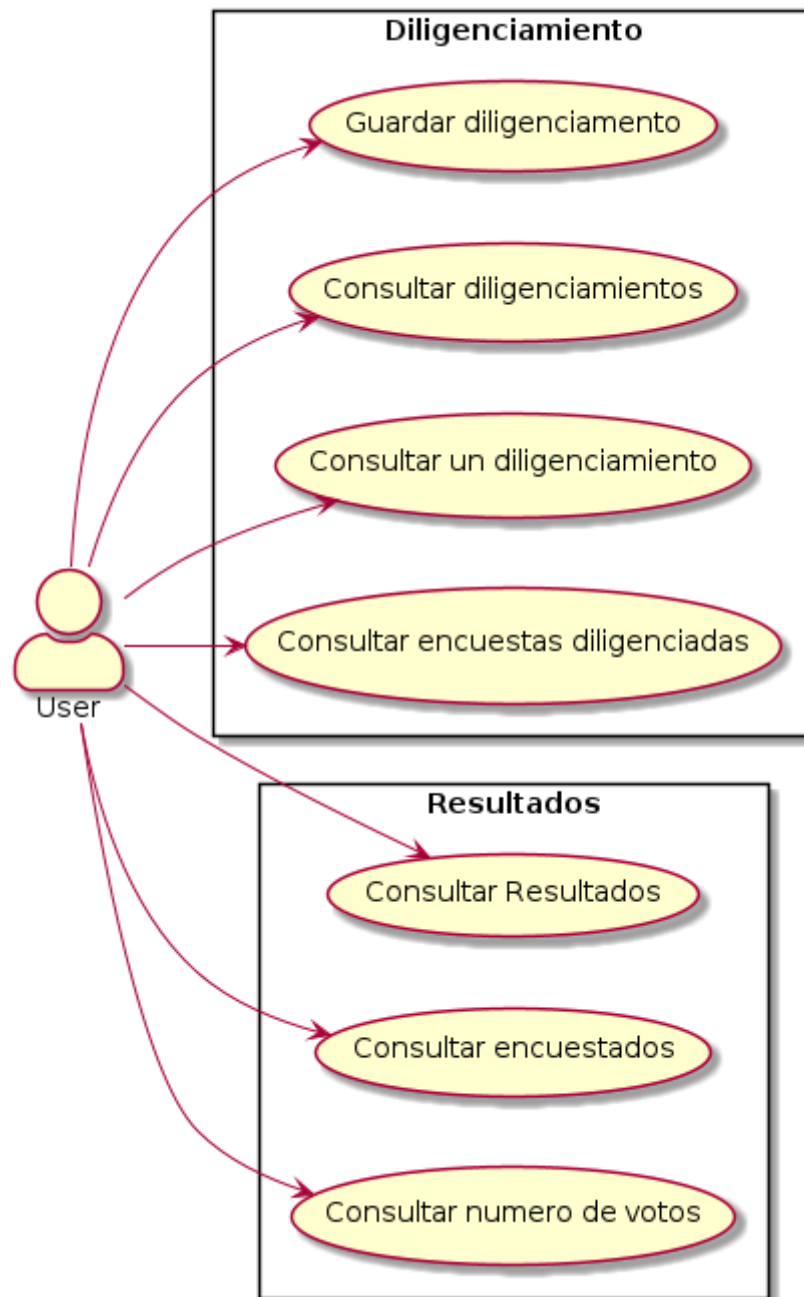


Figura 15. Casos de uso Diligenciamiento

- Módulo de formulador: Encargado de la configuración y cálculo de ecuaciones de indicadores, es decir; contiene los métodos de registro, lectura y eliminación de las configuraciones de fórmulas, como también el cálculo de estas.

Diagrama UML de Casos de Usos:

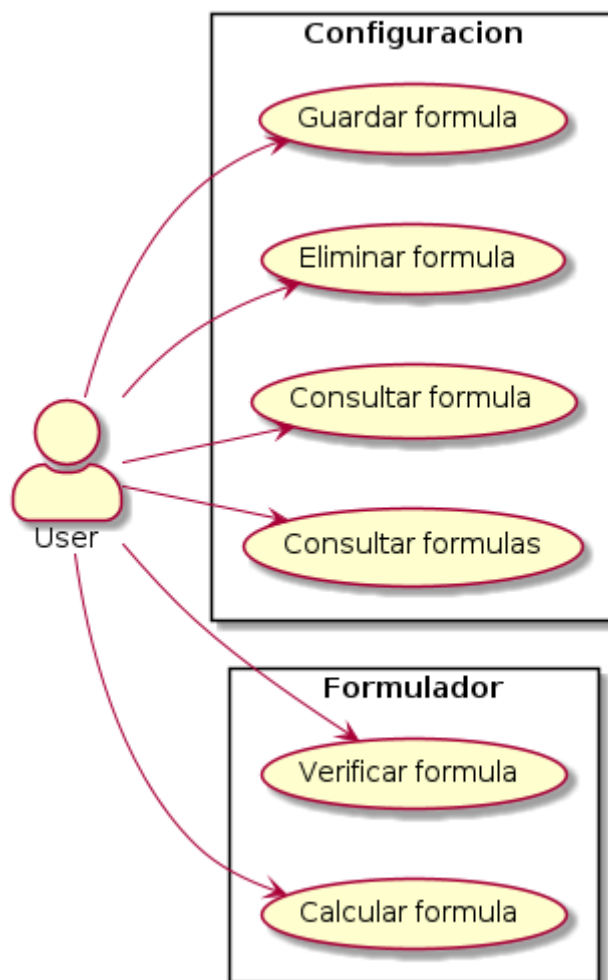


Figura 16. Casos de uso Formulador

- Módulo de indicadores: Encargado de la administración de los indicadores, encuestas y todo lo relacionado a estas.

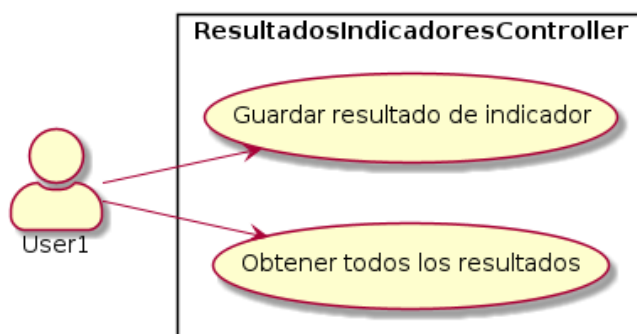


Figura 17. Casos de uso resultados

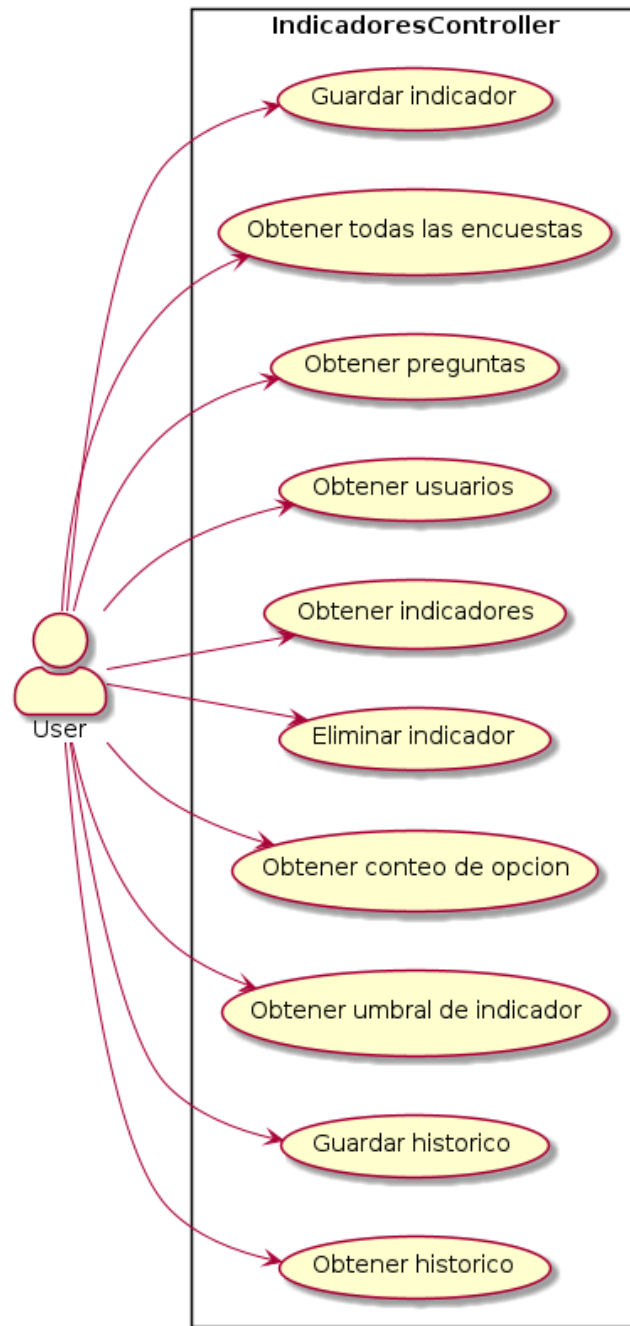


Figura 18. Casos de uso indicadores

- Módulo de análisis: Encargado de las operaciones básicas de actualización, creación, eliminación y consulta de la información de propuestas, propuestas analizadas, demandas y recursos.

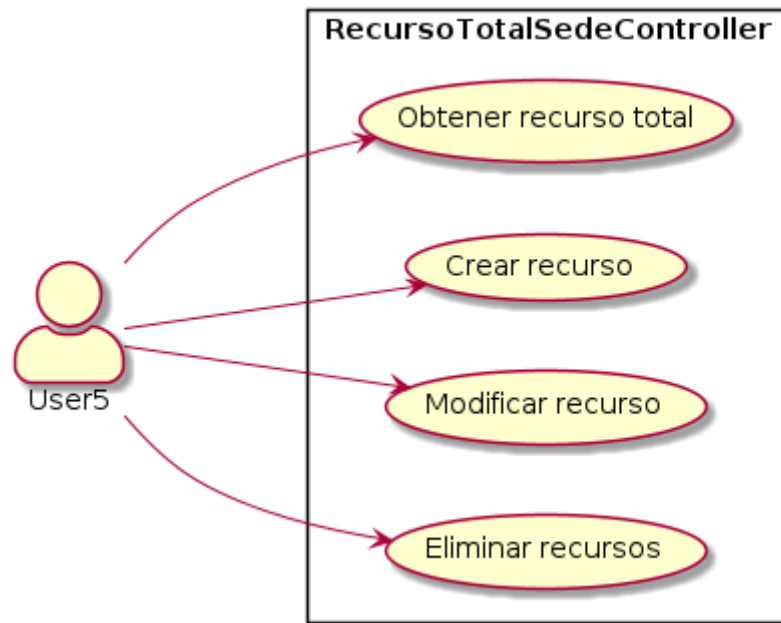


Figura 19. Casos de uso RecursoTotalSede

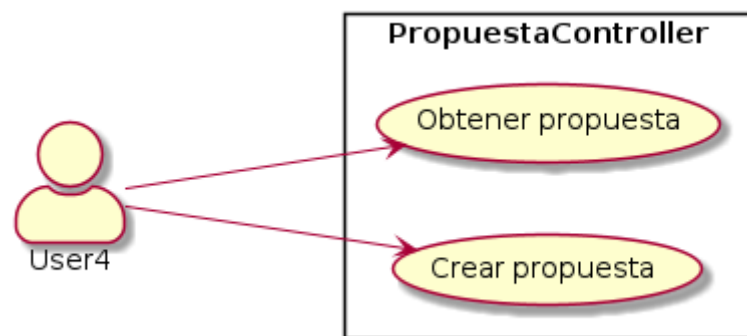


Figura 20. Casos de uso Propuesta

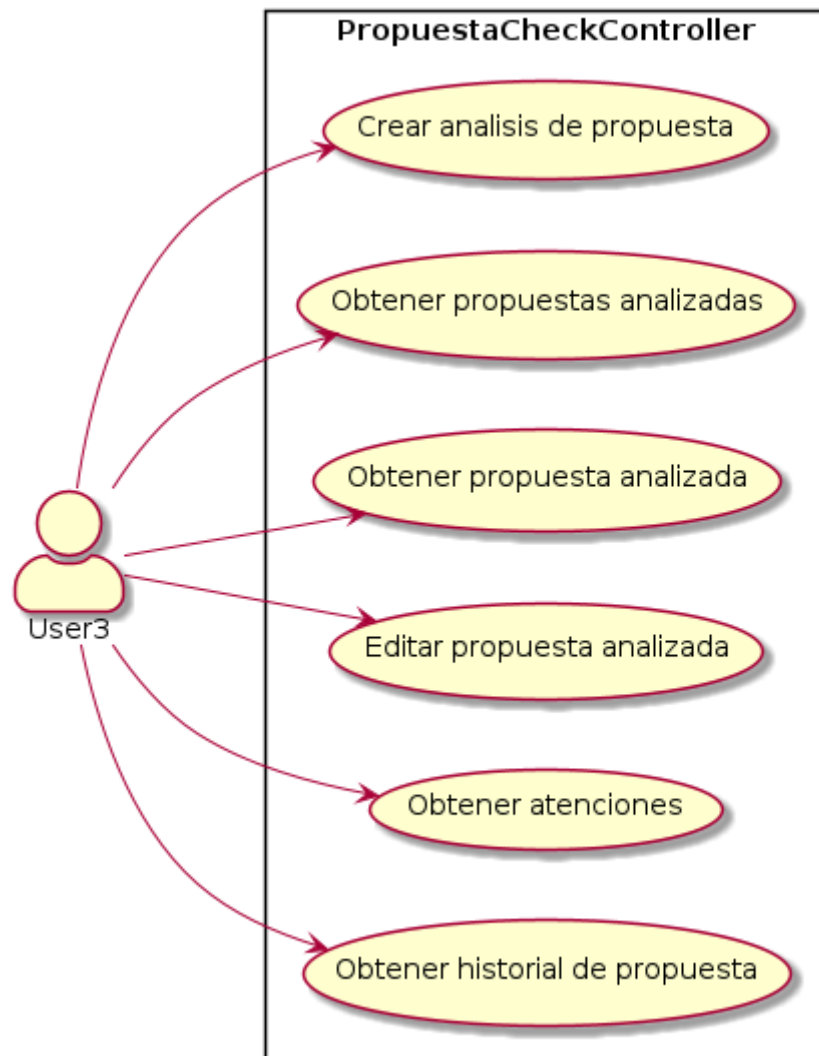


Figura 21. Casos de uso PropuestaCheck

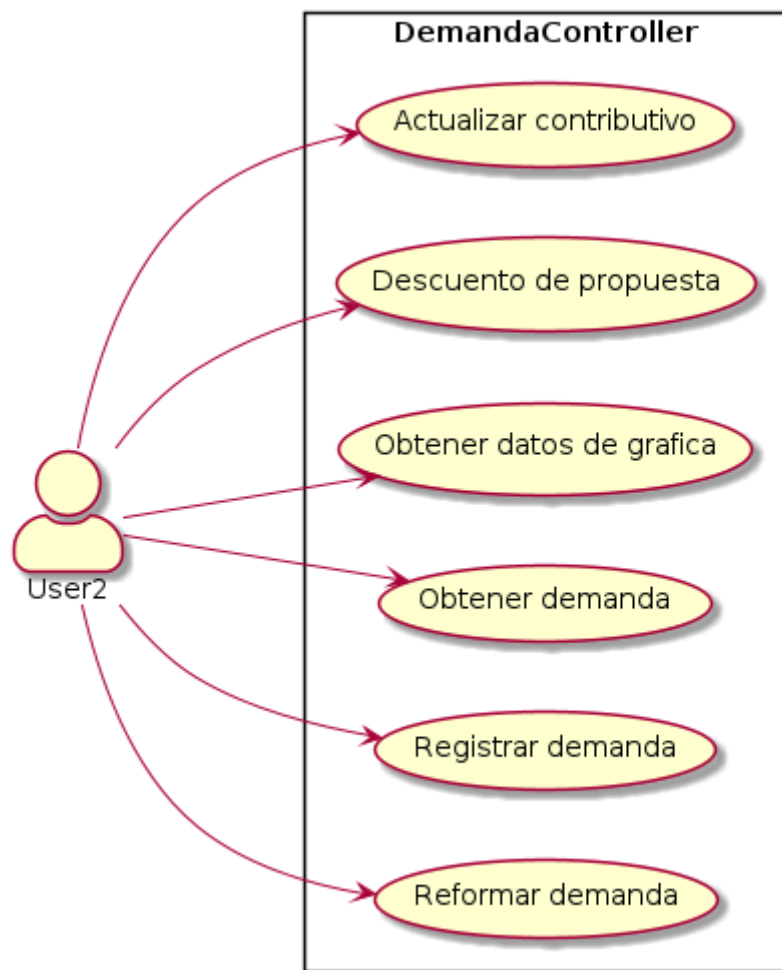


Figura 22. Casos de uso Demanda

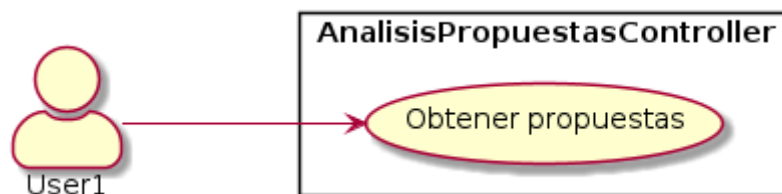


Figura 23. Casos de uso AnalisisPropuesta

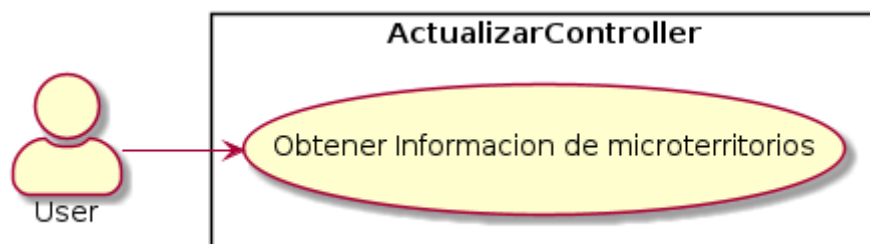


Figura 24. Casos de uso Actualizar

- Módulo de ips: Administración de la información relacionada a la IPS, como RIAS, usuarios, pacientes, componentes y recursos.

Diagrama UML de Casos de Usos:

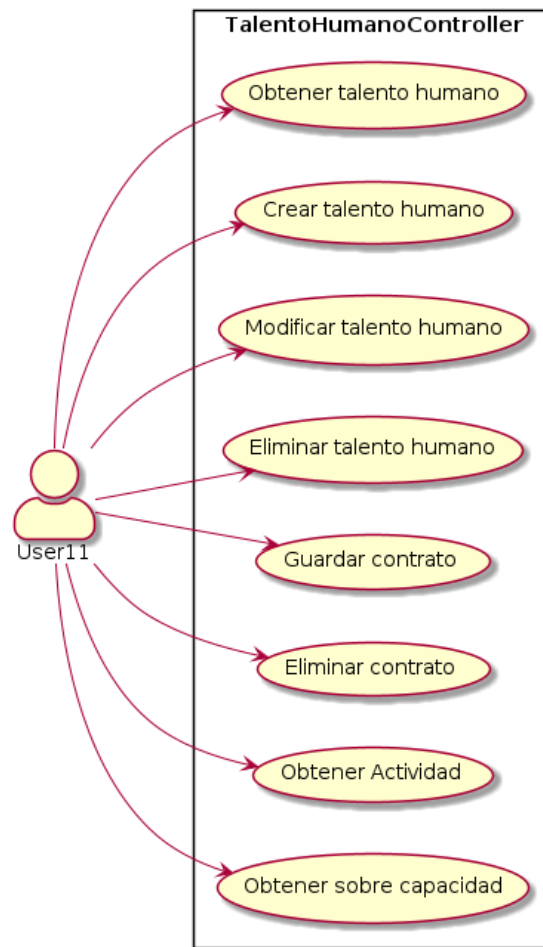


Figura 25. Casos de uso TalentoHumanoControler

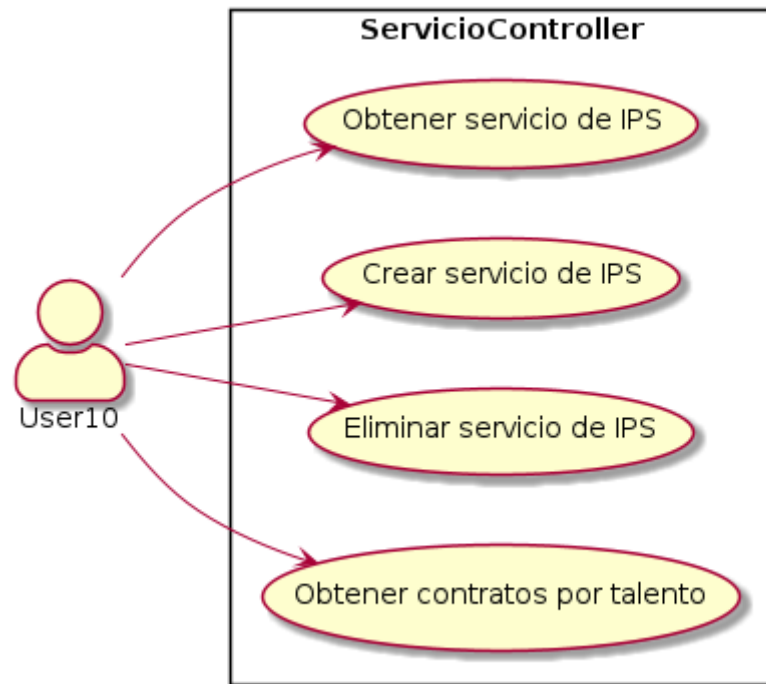


Figura 26. Casos de uso Servicio

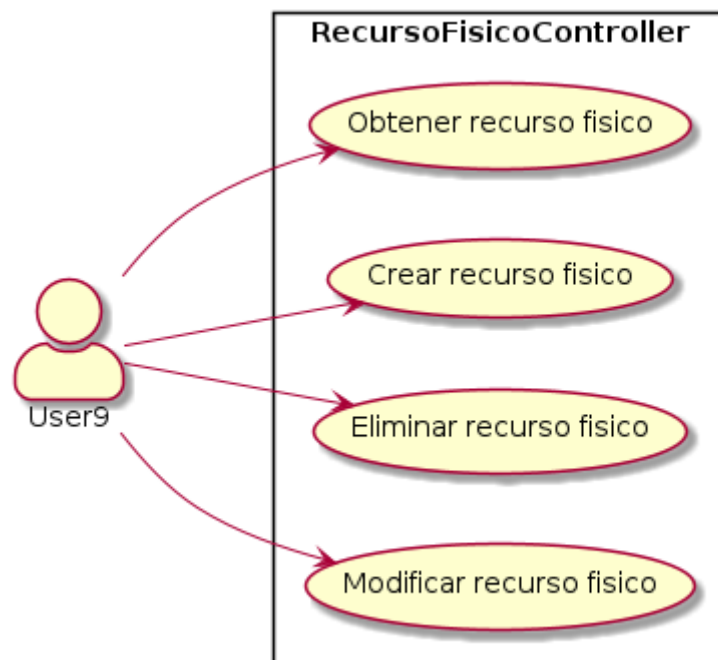


Figura 27. Casos de uso RecursoFisico

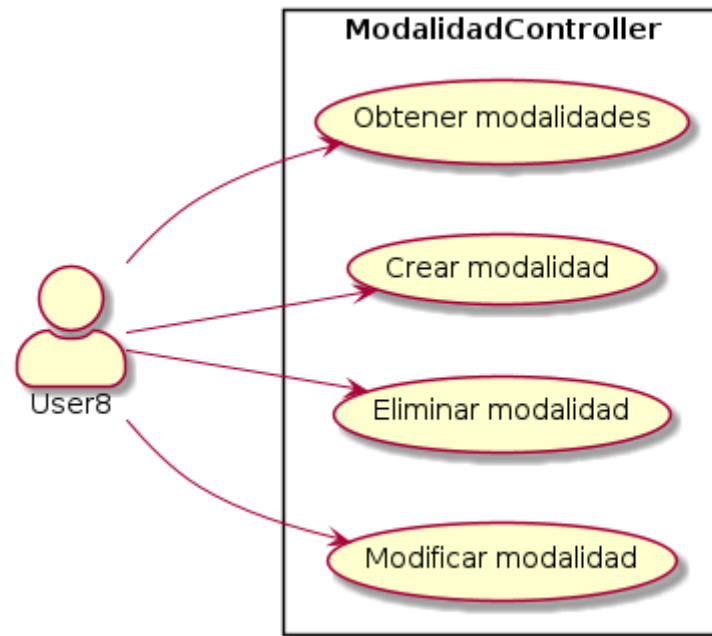


Figura 28. Casos de uso Modalidad

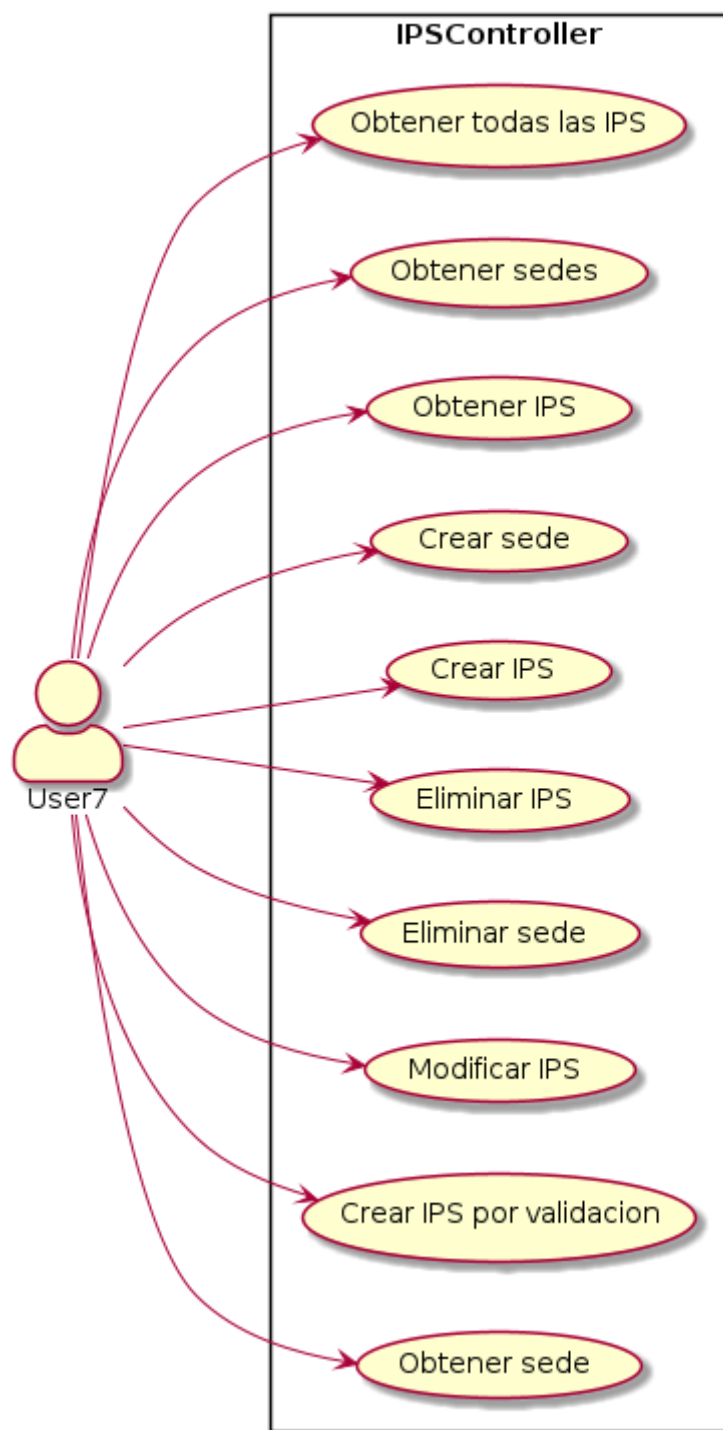


Figura 29. Casos de uso IPS

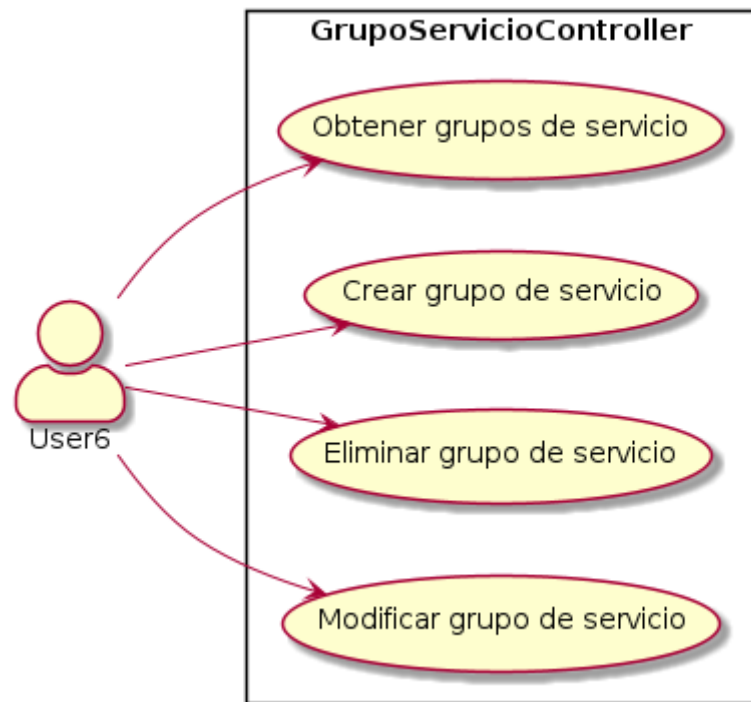


Figura 30. Casos de uso IPS

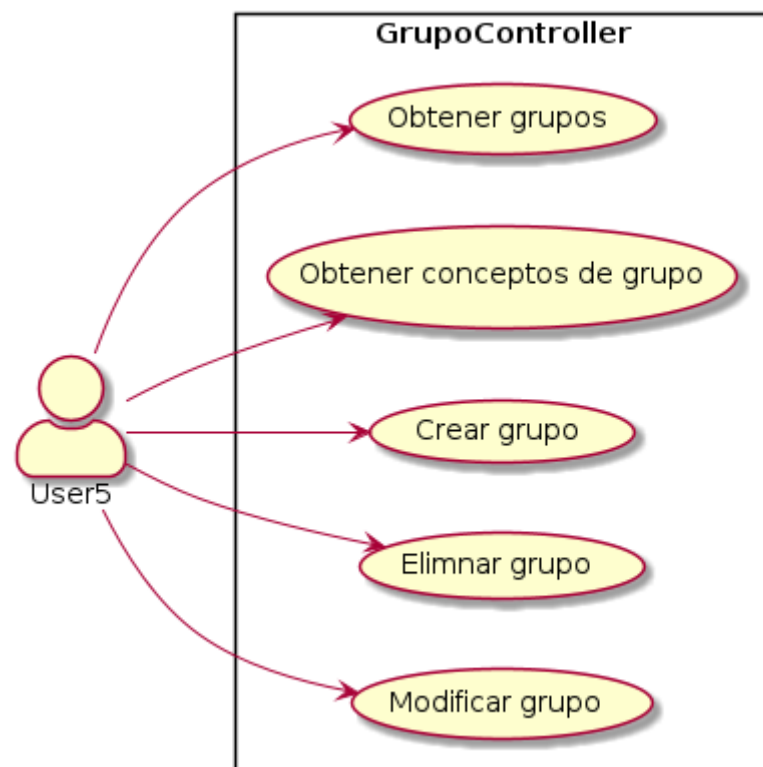


Figura 31. Casos de uso Grupo

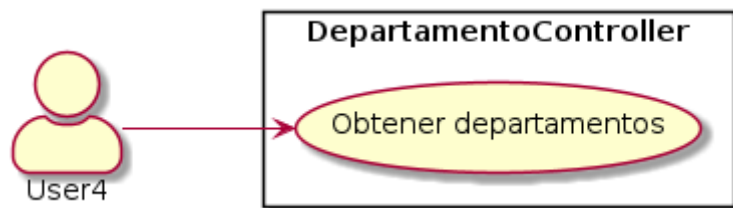


Figura 32. Casos de uso Departamento

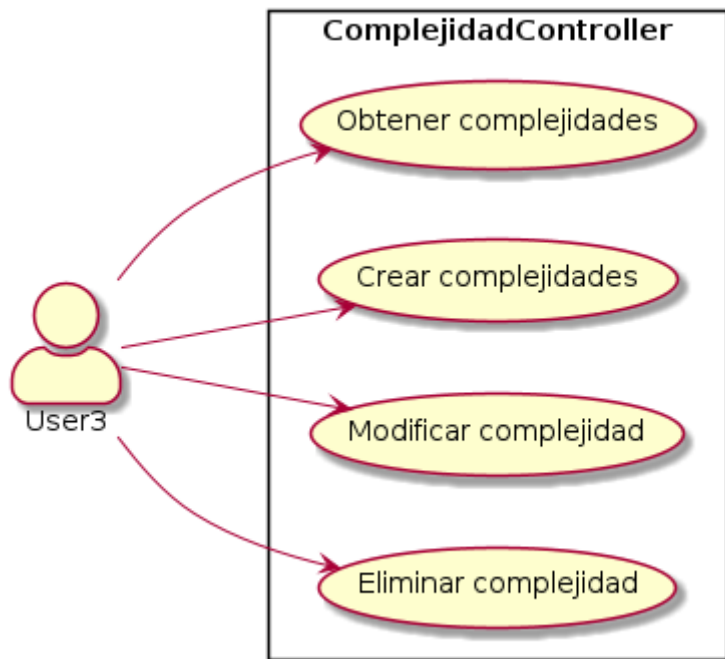


Figura 33. Casos de uso Complejidad

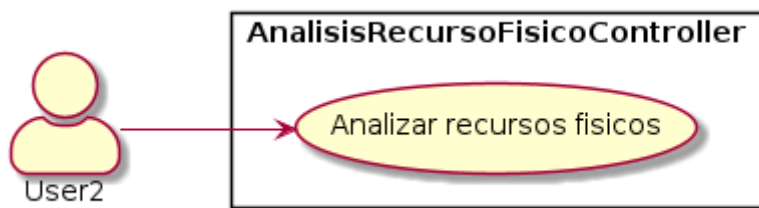


Figura 34. Casos de uso Análisis

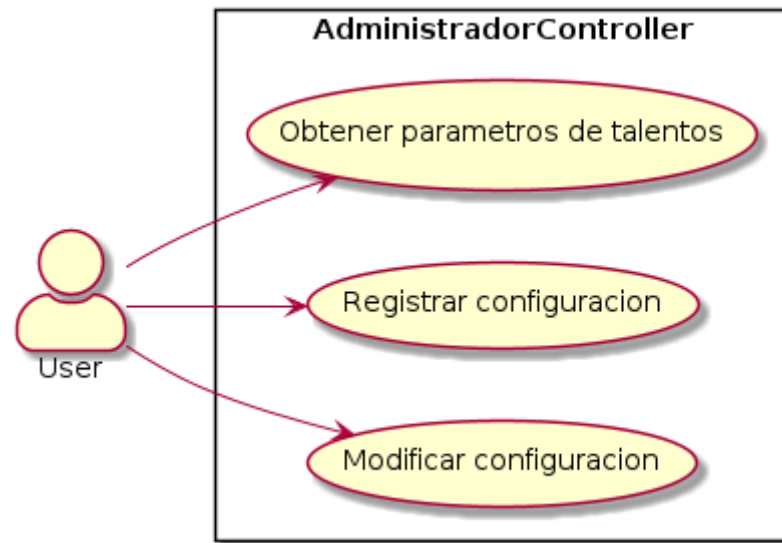


Figura 35. Casos de uso Administrador

- Módulo de microterritorios: Encargado del servicio de ips, sedes, administrador, departamentos, complejidades, servicios, grupos, recursos físicos y talento humano.

Diagrama UML de Casos de Usos:

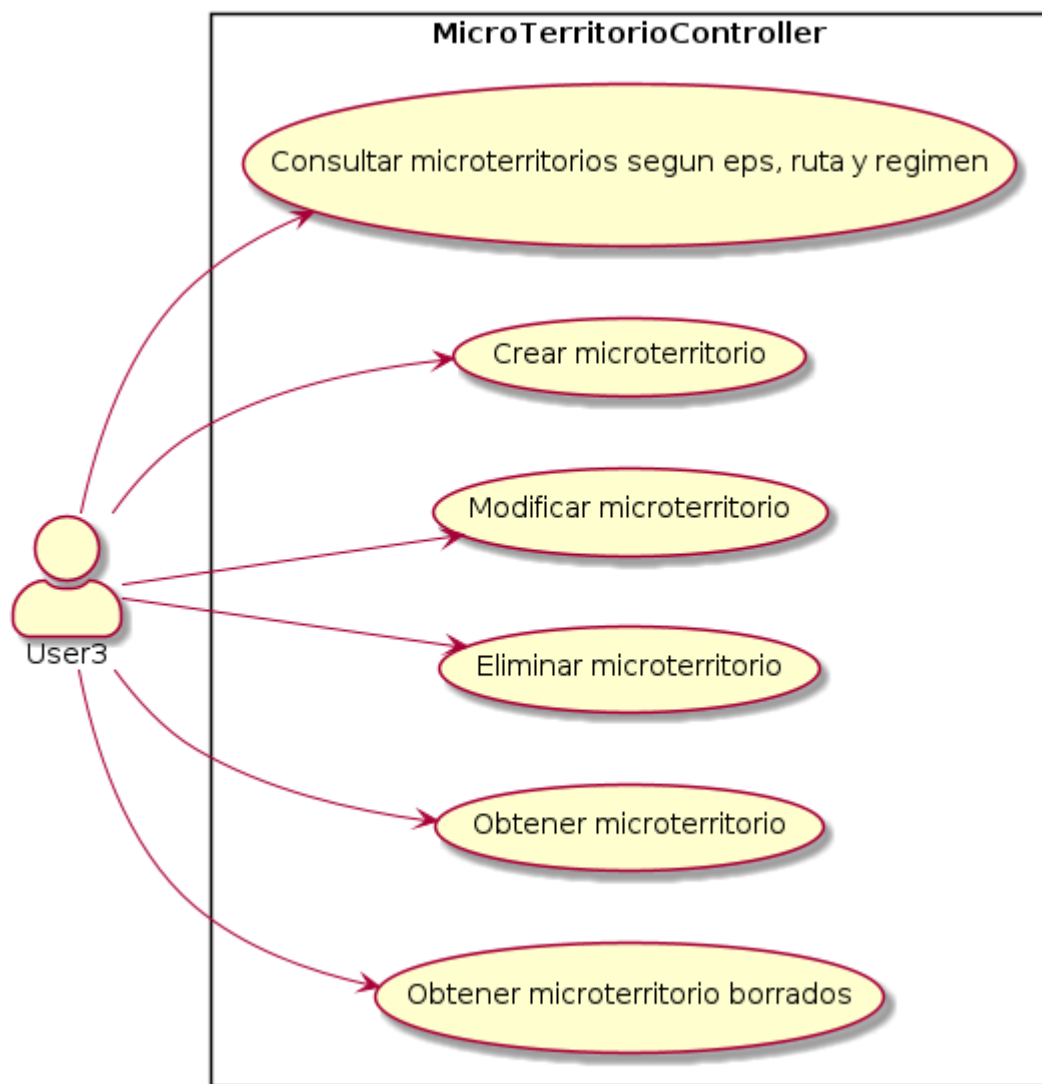


Figura 36. Casos de uso Microterritorios

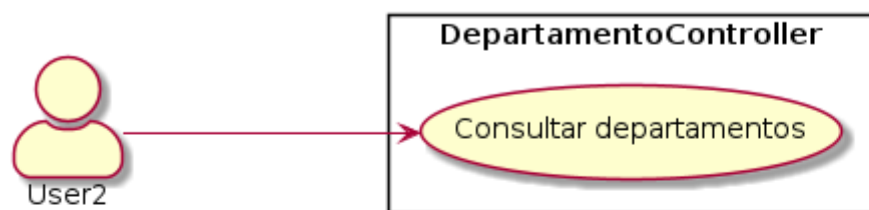


Figura 37. Casos de uso Departamento

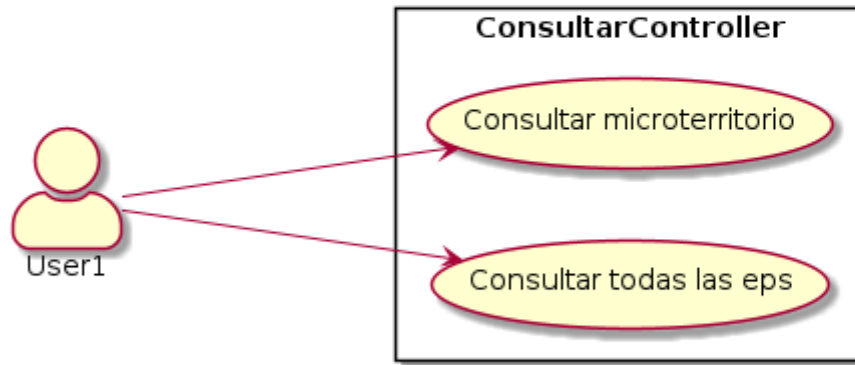


Figura 38. Casos de uso Consultar

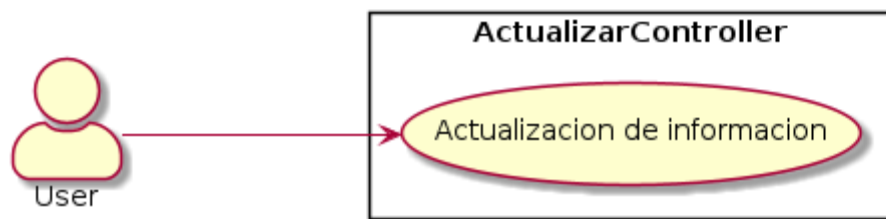


Figura 39. Casos de uso Actualizar

- Módulo de ejecución: Consiste en la administración de los historiales de pacientes en las IPS con sus entradas, traslados y actividades dentro del sistema.

Diagrama UML de Casos de Usos:

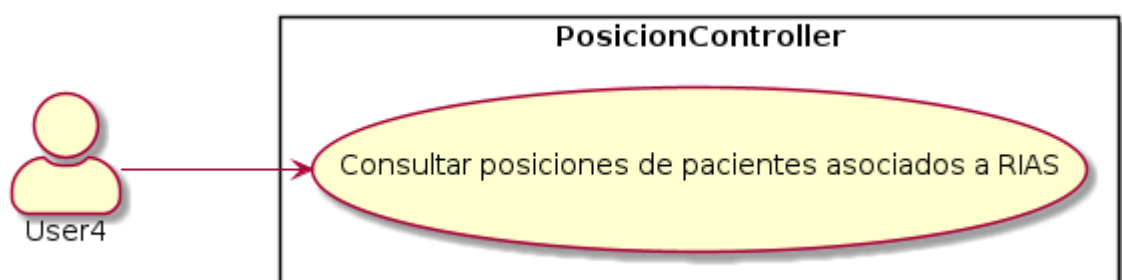


Figura 40. Casos de uso Posición

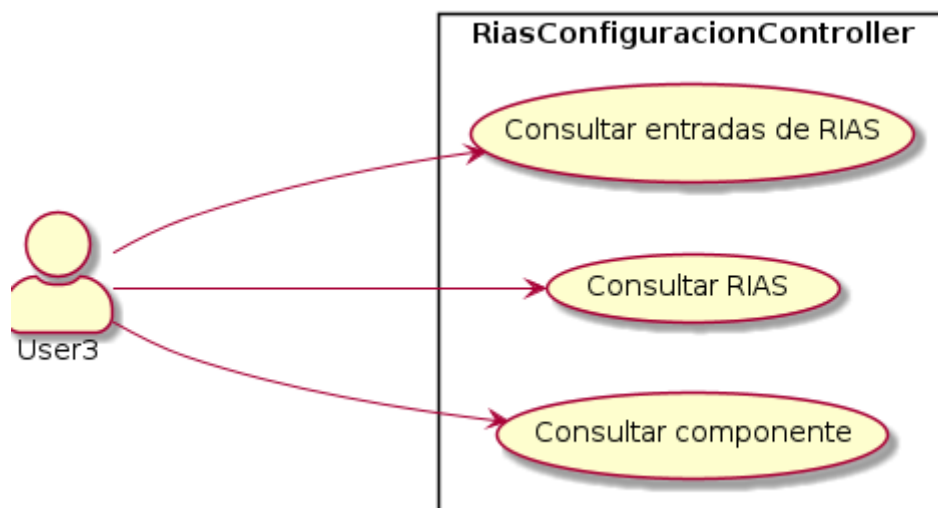


Figura 41. Casos de uso RiasConfiguracion

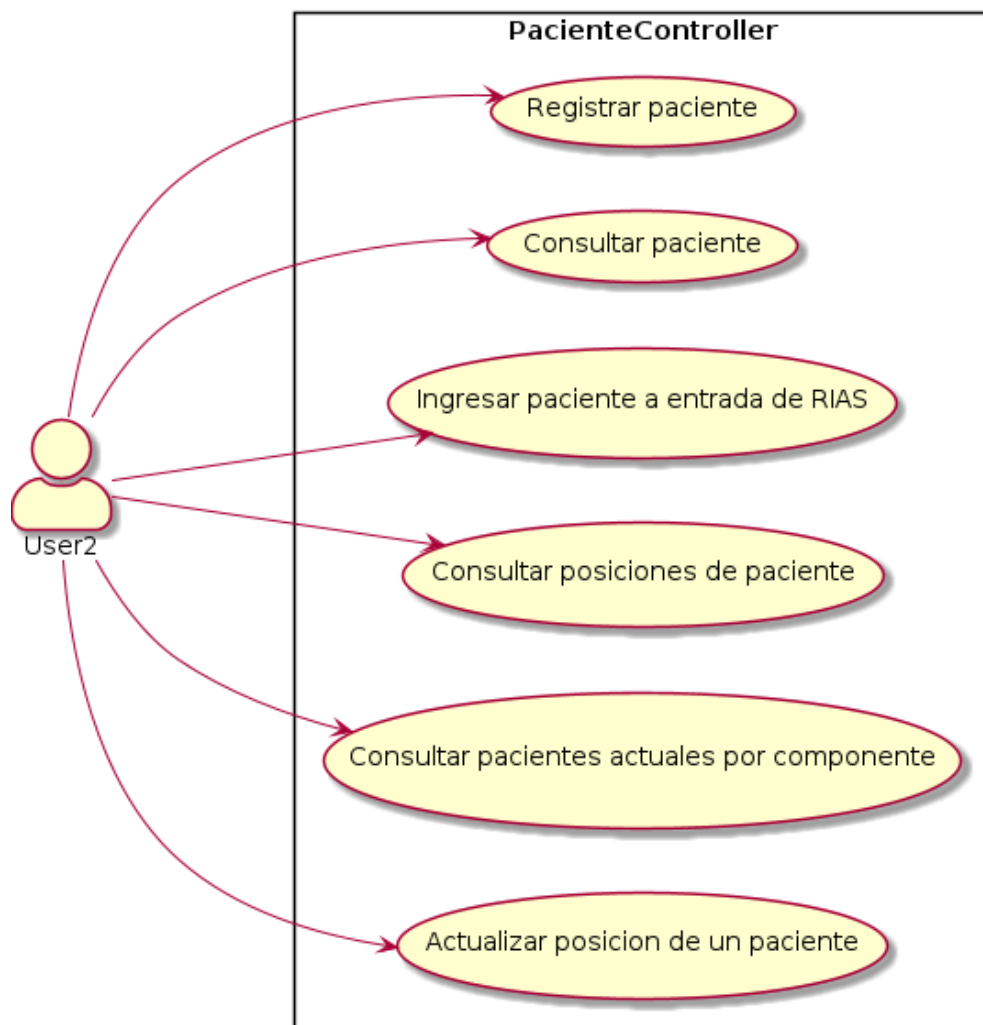


Figura 42. Casos de uso Paciente

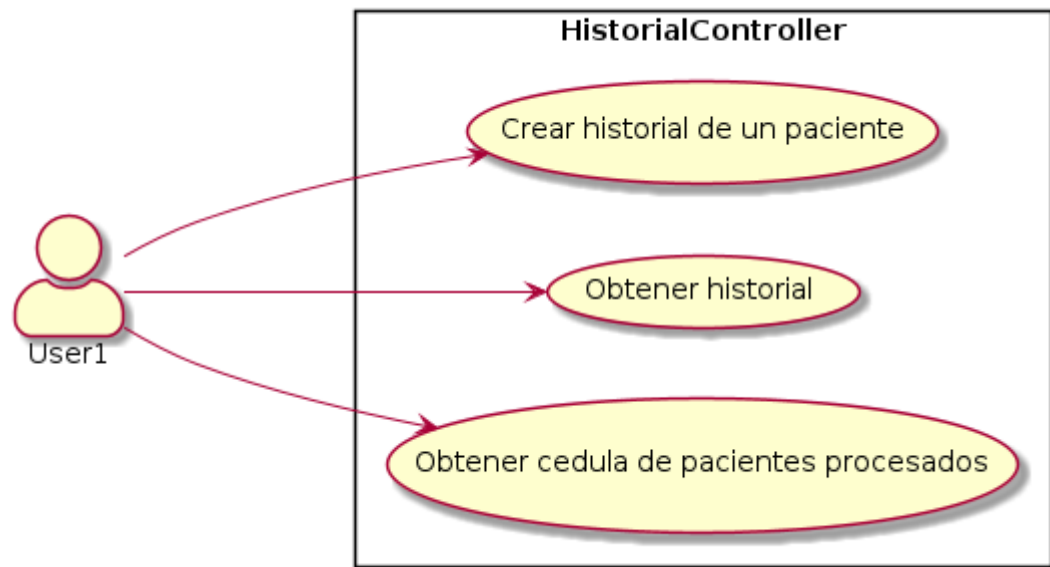


Figura 43. Casos de uso Historial

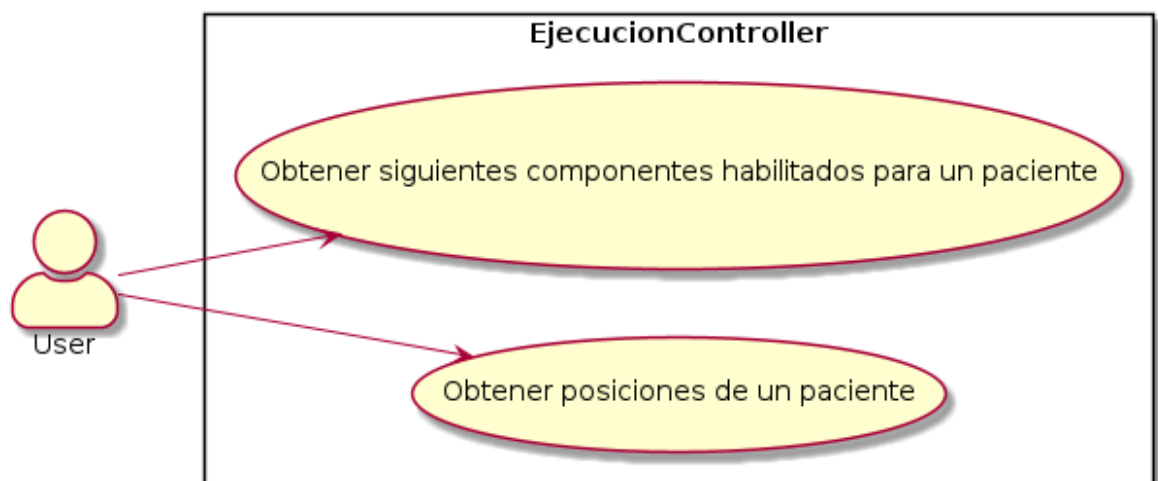


Figura 44. Casos de uso Ejecución

- Módulo de monitoreo : Servicio de monitoreo a pacientes con los componentes de salud.

Diagrama UML de Casos de Usos:

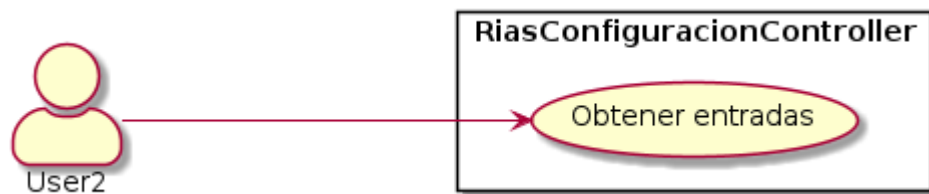


Figura 45. Casos de uso RiasConfiguracion

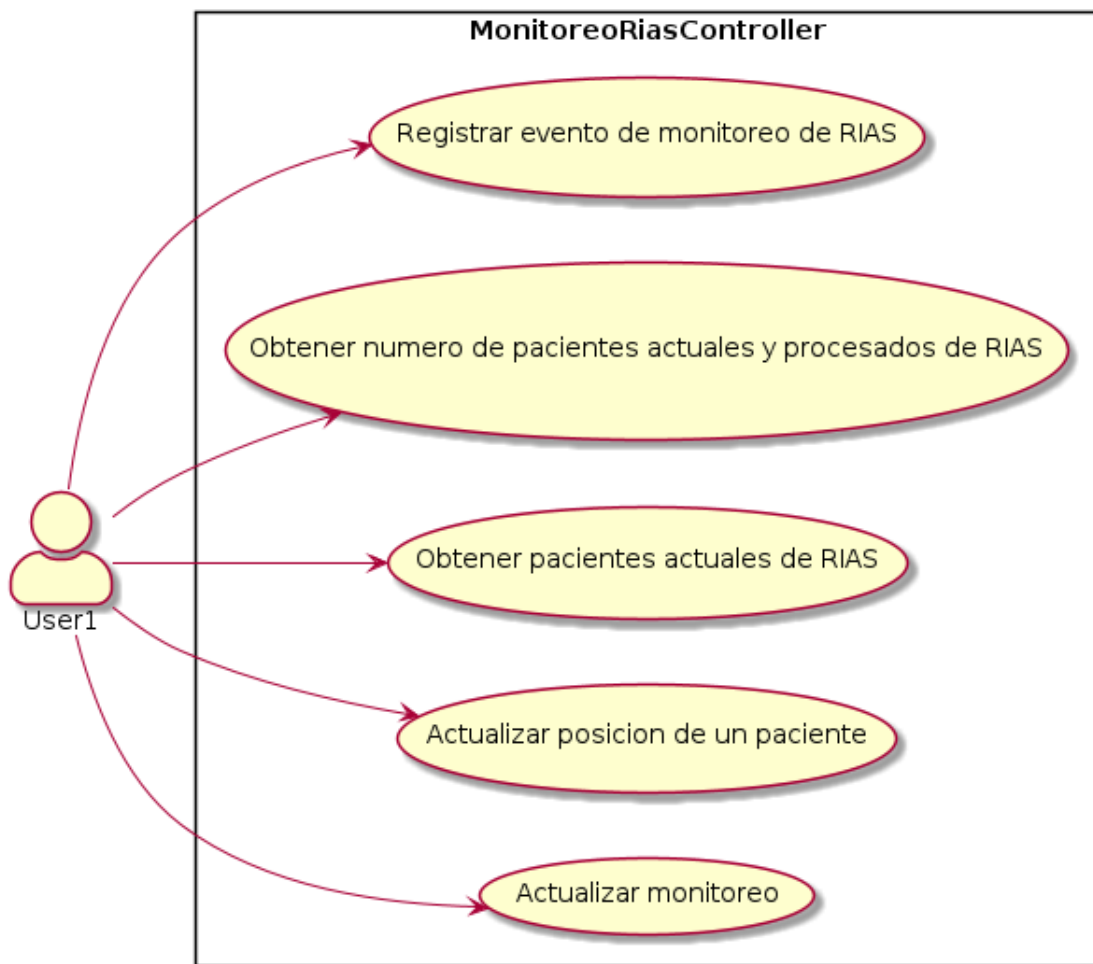


Figura 46. Casos de uso MonitoreoRias

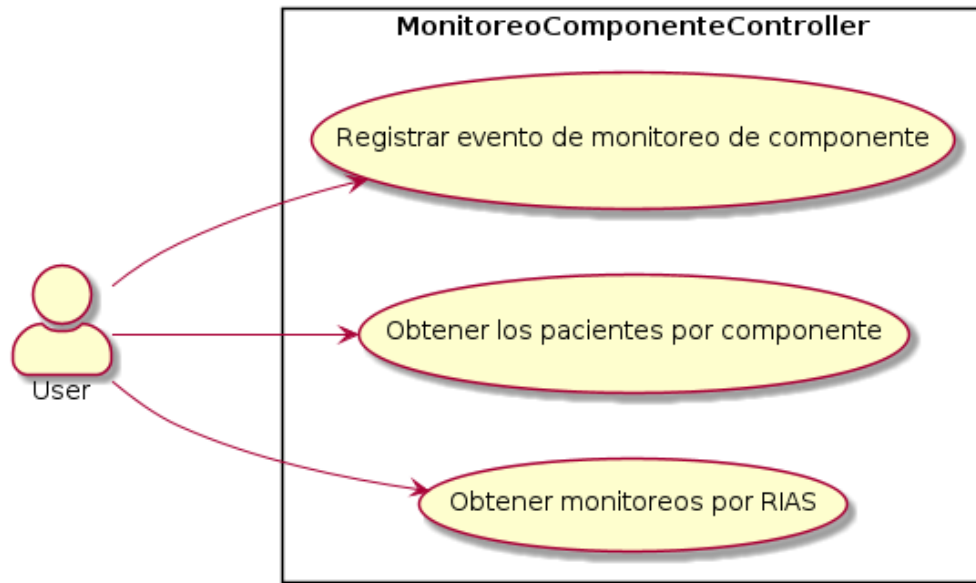


Figura 47. Casos de uso MonitoreoComponente

Para lograr lo anterior, se hicieron diferentes cursos de inducción en Git, Github, Markdown y Plantuml, esto con el fin de tener un aprendizaje previo al momento de ejercer las diferentes tareas asignadas por el director del equipo de desarrollo.

8. Bibliografía

JADCODE. (2015), *¿FRAMEWORK EN EL DESARROLLO DEL SOFTWARE ¿QUÉ SON Y PARA QUÉ PUEDE SER NOS ÚTIL?* Recuperado de <https://jadcode.wordpress.com/2015/03/07/framework-en-el-desarrollo-del-software-que-son-y-para-que-puede-ser-nos-util-2/>

IONOS. (2018), *UML, lenguaje de modelado gráfico* Recuperado de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/uml-lenguaje-unificado-de-modelado-orientado-a-objetos/>